[写文章](https://zhuanlan.zhihu.com/write)

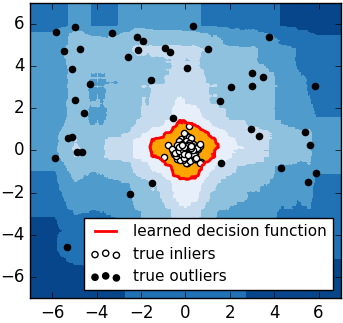
**iForest （Isolation Forest）孤立森林 异常检测 入门篇**

[YeZhu(祝烨)](https://www.zhihu.com/people/yalezhu)[YeZhu(祝烨)](https://www.zhihu.com/people/yalezhu)

6 个月前

iForest （Isolation Forest）孤立森林 是一个基于Ensemble的快速异常检测方法，具有线性时间复杂度和高精准度，是符合大数据处理要求的state-of-the-art算法（详见新版教材“[Outlier Analysis](http://link.zhihu.com/?target=http%3A//link.springer.com/book/10.1007/978-3-319-47578-3)”第5和第6章）。其可以用于网络安全中的攻击检测，金融交易欺诈检测，疾病侦测，和噪声数据过滤等。本文将通俗解释实现方法和日常运用，即无需深厚的数学功底。

首先，我们先了解下该算法的动机。目前学术界对异常（[anomaly detection](http://link.zhihu.com/?target=http%3A//cucis.ece.northwestern.edu/projects/DMS/publications/AnomalyDetection.pdf" \t "_blank)）的定义有很多种，iForest 适用与连续数据（Continuous numerical data）的异常检测，将异常定义为“容易被孤立的离群点 (more likely to be separated)”——可以理解为分布稀疏且离密度高的群体较远的点。用统计学来解释，在数据空间里面，分布稀疏的区域表示数据发生在此区域的概率很低，因而可以认为落在这些区域里的数据是异常的。一个例子如下（[来源](http://link.zhihu.com/?target=http%3A//scikit-learn.org/stable/auto_examples/covariance/plot_outlier_detection.html%23sphx-glr-auto-examples-covariance-plot-outlier-detection-py" \t "_blank)）：



iForest属于Non-parametric和unsupervised的方法，即不用定义数学模型也不需要有标记的训练。对于如何查找哪些点是否容易被孤立（isolated），iForest使用了一套非常高效的策略。假设我们用一个随机超平面来切割（split）数据空间（data space）, 切一次可以生成两个子空间（想象拿刀切蛋糕一分为二）。之后我们再继续用一个随机超平面来切割每个子空间，循环下去，直到每子空间里面只有一个数据点为止。直观上来讲，我们可以发现那些密度很高的簇是可以被切很多次才会停止切割，但是那些密度很低的点很容易很早的就停到一个子空间了。上图里面黑色的点就很容易被切几次就停到一个子空间，而白色点聚集的地方可以切很多次才停止。

怎么来切这个数据空间是iForest的设计核心思想，本文仅介绍最基本的方法。由于切割是随机的，所以需要用ensemble的方法来得到一个收敛值（蒙特卡洛方法），即反复从头开始切，然后平均每次切的结果。iForest 由t个iTree（Isolation Tree）孤立树 组成，每个iTree是一个二叉树结构，其实现步骤如下：

1. 从训练数据中随机选择Ψ个点样本点作为subsample，放入树的根节点。

2. 随机指定一个维度（attribute），在当前节点数据中随机产生一个切割点p——切割点产生于当前节点数据中指定维度的最大值和最小值之间。

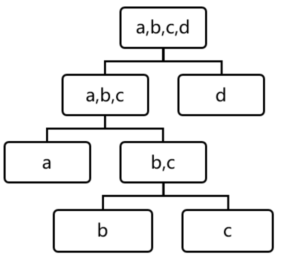
3. 以此切割点生成了一个超平面，然后将当前节点数据空间划分为2个子空间：把指定维度里小于p的数据放在当前节点的左孩子，把大于等于p的数据放在当前节点的右孩子。

4. 在孩子节点中递归步骤2和3，不断构造新的孩子节点，直到 孩子节点中只有一个数据（无法再继续切割） 或 孩子节点已到达限定高度 。

获得t个iTree之后，iForest 训练就结束，然后我们可以用生成的iForest来评估测试数据了。对于一个训练数据x，我们令其遍历每一棵iTree，然后计算x最终落在每个树第几层（x在树的高度）。然后我们可以得出x在每棵树的高度平均值，即 the average path length over t iTrees。\*值得注意的是，如果x落在一个节点中含多个训练数据，可以使用一个公式来修正x的高度计算，详细公式推导见[原论文](http://link.zhihu.com/?target=http%3A//cs.nju.edu.cn/zhouzh/zhouzh.files/publication/icdm08b.pdf" \t "_blank)。

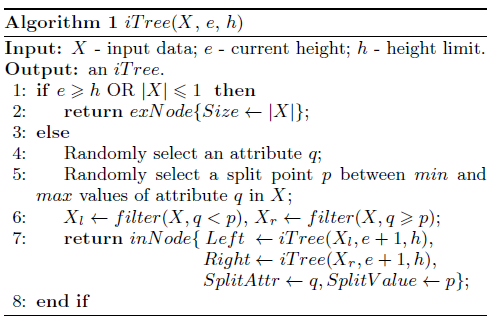
获得每个测试数据的average path length后，我们可以设置一个阈值（边界值），average path length 低于此阈值的测试数据即为异常。也就是说 “iForest identifies anomalies as instances having the shortest average path lengths in a dataset ”(异常在这些树中只有很短的平均高度). \*值得注意的是，论文中对树的高度做了归一化，并得出一个0到1的数值，即越短的高度越接近1（异常的可能性越高）。

4个测试样本遍历一棵iTree的例子如下：

  
  
b和c的高度为3，a的高度是2，d的高度是1。

可以看到d最有可能是异常，因为其最早就被孤立（isolated）了。

生成一棵iTree的详细算法（[来源](http://link.zhihu.com/?target=http%3A//dl.acm.org/citation.cfm%3Fid%3D2939779" \t "_blank)）:

  
X为独立抽取的训练样本。参数e的初始值为0。h是树可以生成的最大高度。

iForest算法默认参数设置如下：

subsample size: 256

Tree height： 8

Number of trees: 100

通俗解释就是——建100棵iTree，每棵iTree最高8层，且每棵iTree都是独立随机选择256个数据样本建成。

**个人见解：**

1. iForest具有线性时间复杂度。因为是ensemble的方法，所以可以用在含有海量数据的数据集上面。通常树的数量越多，算法越稳定。由于每棵树都是互相独立生成的，因此可以部署在大规模分布式系统上来加速运算。

2. iForest不适用于特别高维的数据。由于每次切数据空间都是随机选取一个维度，建完树后仍然有大量的维度信息没有被使用，导致算法可靠性降低。高维空间还可能存在大量噪音维度或无关维度（irrelevant attributes），影响树的构建。对这类数据，建议使用子空间异常检测（Subspace Anomaly Detection）技术。此外，切割平面默认是axis-parallel的，也可以随机生成各种角度的切割平面，详见“[On Detecting Clustered Anomalies Using SCiForest](http://link.zhihu.com/?target=http%3A//link.springer.com/chapter/10.1007/978-3-642-15883-4_18)”。

3. iForest仅对Global Anomaly 敏感，即全局稀疏点敏感，不擅长处理局部的相对稀疏点 （Local Anomaly）。目前已有改进方法发表于PAKDD，详见“[Improving iForest with Relative Mass](http://link.zhihu.com/?target=http%3A//link.springer.com/chapter/10.1007/978-3-319-06605-9_42)”。

4. iForest推动了重心估计（Mass Estimation）理论发展，目前在分类聚类和异常检测中都取得显著效果，发表于各大顶级数据挖掘会议和期刊（如SIGKDD，ICDM，ECML）。

**参考文献：**

iForest 是[刘飞](http://link.zhihu.com/?target=https%3A//feitonyliu.wordpress.com/about/" \t "_blank)博士(Fei Tony Liu)在莫纳什大学就读期间由[陈开明](http://link.zhihu.com/?target=https%3A//federation.edu.au/faculties-and-schools/faculty-of-science-and-technology/staff-profiles/information-technology/kai-ming-ting" \t "_blank)(Kai-Ming Ting)教授和[周志华](http://link.zhihu.com/?target=http%3A//cs.nju.edu.cn/zhouzh/" \t "_blank)(Zhi-Hua Zhou)教授指导发表的。第一个版本是在2008年ICDM上，获得年度最佳论文，扩充版本发表与TKDD。

Liu, Fei Tony, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation forest."*Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*. IEEE, 2008.

Liu, Fei Tony, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation-based anomaly detection."*ACM Transactions on Knowledge Discovery from Data (TKDD)*6.1 (2012): 3.

**论文下载：**

[http://cs.nju.edu.cn/zhouzh/zhouzh.files/publication/icdm08b.pdf](http://link.zhihu.com/?target=http%3A//cs.nju.edu.cn/zhouzh/zhouzh.files/publication/icdm08b.pdf)

[http://cs.nju.edu.cn/zhouzh/zhouzh.files/publication/tkdd11.pdf](http://link.zhihu.com/?target=http%3A//cs.nju.edu.cn/zhouzh/zhouzh.files/publication/tkdd11.pdf)

**源码下载：**

R语言 [Isolation Forest](http://link.zhihu.com/?target=https%3A//sourceforge.net/projects/iforest/)

Python语言 [sklearn.ensemble.IsolationForest - scikit-learn 0.18.1 documentation](http://link.zhihu.com/?target=http%3A//scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html)

Java语言 [Waikato Environment for Knowledge Analysis (WEKA)](http://link.zhihu.com/?target=http%3A//weka.sourceforge.net/packageMetaData/isolationForest/index.html)

Matlab语言 [https://github.com/zhuyemvp/iForest](http://link.zhihu.com/?target=https%3A//github.com/zhuyemvp/iForest)

***全文完，转载必须注明出处****： ©* [*Ye Zhu*](http://link.zhihu.com/?target=http%3A//www.yezhu.com.au/) *2017*